Fast multipole method applied to 3D frequency domain elastodynamics [1]

J.A. Sanz [a], M. Bonnet [b,*], J. Dominguez [a]

[a] *Escuela Superior de Ingenieros, Universidad de Sevilla, Camino de los Descubrimientos s/n, 41092 Sevilla, Spain*

[b] *Solid Mechanics Laboratory (CNRS UMR 7649), Department of Mechanics, École Polytechnique, 91128 Palaiseau cedex, France*

Abstract

This article is concerned with the formulation and implementation of a fast multipole-accelerated BEM for 3-D elastodynamics in the frequency domain, based on the so-called diagonal form for the expansion of the elastodynamic fundamental solution, a multi-level strategy. As usual with the FM-BEM, the linear system of BEM equations is solved by GMRES, and the matrix is never explicitly formed. The truncation parameter in the multipole expansion is adjusted to the level, a feature known from recent published studies for the Maxwell equations. A preconditioning strategy based on the concept of sparse approximate inverse (SPAI) is presented and implemented. The proposed formulation is assessed on numerical examples involving $O(10^5)$ BEM unknowns, which show in particular that, as expected, the proposed FM-BEM is much faster than the traditional BEM, and that the GMRES iteration count is significantly reduced when the SPAI preconditioner is used.

Key words: Fast Multipole Method, Boundary Element Method, 3D Elastodynamics.

1 INTRODUCTION

The boundary element method (BEM) is a mesh reduction method. However, in traditional BEM implementations, the resulting dimensional advantage with

* Corresponding author. Tel.: +33 6933 5746; fax: +33 6933 3026
Email address: bonnet@lms.polytechnique.fr (M. Bonnet).

[1] *Engng. Anal. Bound. Elem..* **32**:787–795 (2008)

respect to domain discretization methods is offset by the fully-populated nature of the governing discretized system of equations, with set-up and solution times rapidly increasing with the number N of unknown BEM degrees of freedon (DOFs). It is therefore important to formulate alternative strategies which allow to take advantage of the proven efficiency of BEMs for e.g. analyses involving unbounded media, fracture mechanics, or inverse problems when large numbers of degrees of freedom (DOFs) are considered. Approaches such as symmetric Galerkin BEMs and BEM-FEM coupling, when developed within traditional set-up and solution algorithms, do not fully address these concerns because they do not lead to lowered complexities (i.e. CPU and memory requirements grow with N at the same rate).

Fast BEMs, i.e. BEMs having a complexity lower than that of traditional BEMs, appeared around 1985. Rokhlin [1] presented a $O(N)$ per iteration integral-equation approach for solving 2-D Laplace problems. The Fast Multipole Method (FMM) terminology was introduced first by Greengard [2], in the context of many-particle simulations. Thereafter, the FMM led to fast multipole-accelerated boundary element methods (FM-BEMs), whose scope and capabilities have since rapidly progressed, especially in connection with application in electromagnetics [3,4], but also in other fields including computational mechanics. Many of these investigations are summarized in a review article by Nishimura [5].

This article is concerned with the formulation and implementation of a FM-BEM for 3-D elastodynamics in the frequency domain. Very few references address this particular area of application. FM-BEM formulation for elastodynamics in the time domain and the frequency domain have been proposed by Takahashi *et al.* [6] and Fujiwara [7], respectively. This work is based on the same general approach as that of [7], which uses the so-called diagonal form for the expansion of the elastodynamic fundamental solution. This approach is suitable for moderate to medium frequencies. The upper limit stems from the fact that the size N becomes intractable at high frequencies, while the diagonal form breaks down at very low frequencies and must be replaced with other types of expansions, based on e.g. the Wigner–3j symbols [8] or the so-called stable plane wave expansion [9]. The present work differs from [7] in details of the formulation that substantially impact the overall efficiency, and especially the fact that the truncation parameter in the multipole expansion should be adjusted to the subdivision level. This feature, known from previous studies for the Maxwell equations (e.g. [10]), is incorporated in the present implementation but not in that of [7].

This article is organized as follows. The necessary background concepts and notations are reviewed in section 2. Then, some details of the elastodynamic FM-BEM are discussed in section 3. A preconditioning strategy based on the concept of sparse approximate inverse (SPAI) is presented in section 4. Finally,

the present formulation is assessed on numerical examples in section 5, and the paper closes with concluding remarks.

2 PRELIMINARIES

The main idea behind the FMM is to reformulate the fundamental solution (or kernel) in terms of products of functions of the integration and collocation points. Together with the use of iterative techniques for solving linear systems, this provides algorithms of complexity $O(N \log^{\alpha} N)$ per iteration (see [11] for a mathematical justification), where N is the number of BEM DOFs. Such kernel reformulations hold for sets of integration and collocation points each clustered around a origin point (called pole); moreover, the separation distance between the two poles must be such that the two clusters are well-separated. Hence, many such poles must be introduced.

There are basically two kinds of FMMs: the *one-level* FMM and the *multilevel* FMM. Both are based on dividing the spatial region of interest (i.e. that containing the boundary over which the integral equation is formulated) into cubic regions called cells. Pairs of well-separated (i.e. completely disjoint) cells are then used for defining the above-mentioned clusters of integration and collocation points. This article is concerned with the multi-level approach, known to have a lower complexity than the one-level approach, where the spatial region of interest is first divided into $4 \times 4 \times 4$ cubic cells. Each non-empty cell is then divided into $2 \times 2 \times 2$ smaller cells. This subdivision process is repeated until some criterion (such as a preset minimum number of degrees of freedom per cell, or a minimum cell size) is reached.

2.1 Classical BEM formulation

Let Ω denote the (bounded or unbounded) region of space occupied by an elastic medium with isotropic constitutive properties defined by μ (shear modulus), ν (Poisson's ratio) and ρ (mass density). Assuming a time-harmonic motion with circular frequency ω and the absence of body forces, the classical elastodynamic BEM formulation is based on the well-known integral identity (see e.g. [12])

$$C_{ip}(\mathbf{x})u_p(\mathbf{x}) = \int_\Gamma U^\star_{ip}(\mathbf{x},\mathbf{y})t^n_p(\mathbf{y})d\Gamma_y - \int_\Gamma T^{\star n}_{ip}(\mathbf{x},\mathbf{y})u_p(\mathbf{y})d\Gamma_y \qquad (\mathbf{x} \in \Gamma) \quad (1)$$

where t^n_p and u_p, respectively, denote the Cartesian components of the displacements and tractions on the boundary Γ of Ω, C_{ip} are the coefficients of the free term, which depend on the local geometry of Γ at $\mathbf{x}$ and is equal

to $(1/2)\delta_{ij}$ in the usual case where Γ is smooth at $\mathbf{x}$ (with δ_{ij} denoting the Kronecker symbol). In (1) and thereafter, the usual Einstein convention of implicit summation over repeated indices is used. The second integral term in Eq. (1) is a Cauchy principal value singular integral, which can be dealt with by means of either a regularization procedure [13] or a specialized direct quadrature approach [14]. The fundamental solution of 3D frequency domain isotropic elastodynamics is given, in terms of Cartesian components of displacements $U^\star_{ip}$ and tractions $T^{\star n}_{ip}$, by

$$U^\star_{ip}(\mathbf{x},\mathbf{y}) = \frac{1}{4\pi\mu k_S^2}\left(e_{rti}e_{rsp}\frac{\partial}{\partial x_t}\frac{\partial}{\partial y_s}\frac{\exp(\mathrm{i}k_S\,|\mathbf{x}-\mathbf{y}|)}{|\mathbf{x}-\mathbf{y}|} + \frac{\partial}{\partial x_i}\frac{\partial}{\partial y_p}\frac{\exp(\mathrm{i}k_P\,|\mathbf{x}-\mathbf{y}|)}{|\mathbf{x}-\mathbf{y}|}\right) \tag{2}$$

$$T^{\star n}_{ip}(\mathbf{x},\mathbf{y}) = E_{jkpl}\frac{\partial}{\partial y_l}U^\star_{ij}(\mathbf{x},\mathbf{y})n_k(\mathbf{y}) \tag{3}$$

where k_S , k_P are the transversal and longitudinal (S and P) wave numbers, respectively, $\mathrm{i} = \sqrt{-1}$ is the imaginary unit, e_{abc} are the components of the permutation tensor, $\mathbf{n}$ is the unit normal to Γ exterior to Ω, and E_{ijkl} are the components of the isotropic elasticity tensor, i.e.:

$$E_{ijkl} = \mu\left(\frac{2\nu}{1-2\nu}\delta_{ij}\delta_{kl} + \delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) \tag{4}$$

2.2 Multipole expansion of the elastodynamic fundamental solution

The function $\exp(\mathrm{i}kr)/(4\pi r)$ appearing in (2), which is the fundamental solution for the Helmholtz equation with wavenumber k, admits the plane-wave expansion [15]:

$$\begin{aligned}&\frac{\exp(\mathrm{i}k\,|\mathbf{x}-\mathbf{y}|)}{4\pi\,|\mathbf{x}-\mathbf{y}|}\\ &= \frac{\mathrm{i}k}{16\pi^2}\lim_{L\to\infty}\int_S \exp\Big(\mathrm{i}k\langle\mathbf{s},\mathbf{O}_x-\mathbf{x}\rangle\Big)T_L(\mathbf{s};k,\mathbf{O}_y-\mathbf{O}_x)\exp\Big(\mathrm{i}k\langle\mathbf{s},\mathbf{y}-\mathbf{O}_y\rangle\Big)d\mathbf{s}\end{aligned} \tag{5}$$

where $S = \{\mathbf{s}\in\mathbb{R}^3, |\mathbf{s}=1|\}$ denotes the unit sphere, the brackets $\langle\cdot,\cdot\rangle$ indicate the inner (dot) product in $\mathbb{R}^3$, and the *transfer function* T_L is given by

$$T_L(\mathbf{s};k,\mathbf{z}) = \sum_{0\le l\le L}(2l+1)\,\mathrm{i}^l h_l^{(1)}(k\,|\mathbf{z}|)\,P_l(\cos(\mathbf{s},\mathbf{z})) \tag{6}$$

in terms of the spherical Hankel functions of the first kind $h_l^{(1)}$ and the Legendre polynomials P_l. The *poles* $\mathbf{O}_x$ and $\mathbf{O}_y$ must be close to the collocation point $\mathbf{x}$ and the integration point $\mathbf{y}$, respectively.

Then, on substituting (5) and (6) into expressions (2) and (3) of the elastodynamic fundamental solution, the multipole expansion of $U^{\star}_{ip}$ and $T^{\star n}_{ip}$ is readily obtained as

$$U^{\star}_{ip}(\mathbf{x},\mathbf{y}) = \frac{1}{16\pi^2\mu}\sum_{q=1}^{Q}\omega_q \mathrm{i}\Big[k_S D_{ri}(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)T_L(\mathbf{s};k_S,\mathbf{O}_y-\mathbf{O}_x)D_{rp}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q) + \frac{k_P^3}{k_S^2}D_i(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)T_L(\mathbf{s};k_P,\mathbf{O}_y-\mathbf{O}_x)D_p(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)\Big] \tag{7}$$

$$T^{\star n}_{ip}(\mathbf{x},\mathbf{y}) = -\frac{1}{16\pi^2\mu}\sum_{q=1}^{Q}\omega_q E_{jkpl} n_k (s_q)_l \Big[k_S^2 D_{ri}(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)T_L(\mathbf{s};k_S,\mathbf{O}_y-\mathbf{O}_x)D_{rj}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q) + \frac{k_P^4}{k_S^2}D_i(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)T_L(\mathbf{s};k_P,\mathbf{O}_y-\mathbf{O}_x)D_j(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)\Big] \tag{8}$$

where the integration on the unit sphere of Eq. (5) has been replaced with a Q-point numerical quadrature, with $\mathbf{s}_q$ and ω_q denoting the quadrature points on S and the associated weights. Note that Eqs. (7) and (8) feature, through the transfer function T_L, the two wavenumbers k_S and k_P, as expected. Moreover, the functions $D_{ri}(\mathbf{z},\mathbf{s})$ and $D_i(\mathbf{z},\mathbf{s})$ are defined by the formulae

$$D_{ri}(\mathbf{z},\mathbf{s}) \stackrel{\text{def}}{=} e_{rti}(s_q)_t \exp\Big(\mathrm{i}k_S\langle\mathbf{s}_q,\mathbf{z}\rangle\Big)$$
$$D_i(\mathbf{z},\mathbf{s}) \stackrel{\text{def}}{=} (s_q)_i \exp\Big(\mathrm{i}k_P\langle\mathbf{s}_q,\mathbf{z}\rangle\Big)$$

2.3 Numerical quadrature over the unit sphere

On parameterizing unit vectors $\mathbf{s} \in S$ using spherical angular coordinates (θ,ϕ), a frequently chosen [10,16] numerical quadrature scheme is based on $Q = (L+1)(2L+1)$ points $\mathbf{s}_q$ of the form

$$\mathbf{s}_q = \Big(\sin(\theta_a)\cos(\phi_b)\ \ \sin(\theta_a)\sin(\phi_b)\ \ \cos(\theta_a)\Big),$$

where L is the truncation level used in (6), $\cos\theta_a$ $(1 \le a \le L+1)$ are the abscissae for the $(L+1)$-point Gauss-Legendre 1-D quadrature rule over $[-1,1]$ (with θ_a such that $0 < \theta_a < \pi$) and $\phi_b = 2\pi b/(2L+1)$ $(0 \le b \le 2L)$ are uniformly-spaced abscissae on $[0,2\pi]$. The associated weights are $w_q = 2\pi w_a^{\theta}/(2L+1)$, where w_a^{θ} are the Gauss-Legendre weights for the $(L+1)$-point one-dimensional rule. The chosen dependence of Q in L stems from the fact that the transfer function (6) involves Legendre polynomials of degree up to L.

Moreover, it is known [10,16,17] that to ensure a given accuracy in (6), the truncation parameter L must be adjusted to the cell size d. A often-used

empirical formula for that purpose is [10,16]

$$L(k) = \sqrt{3}kd + C\log_{10}\left(\sqrt{3}kd + \pi\right) \tag{9}$$

where the constant C is determined on the basis of numerical experiments ($C = 4 - 8$ being typical values). A similar approach, based on fine-tuning the coefficient C of another empirical truncation formula with a $O((kd)^{1/3})$ term replacing the logarithmic term of (9), is reported in [18]. Using either approach, L is roughly proportional to the cell size. In practice, the truncation parameter in (6) is set according to

$$L_\alpha = \mathrm{Max}(5, L(k_\alpha)) \qquad (\alpha = P, S) \tag{10}$$

depending on whether $k = k_P$ or $k = k_S$ is used in the expansion. On noting that $L_S \geq L_P$, the number of quadrature points Q is set for simplicity to $Q = (L_S + 1)(2L_S + 1)$ in all cases.

3 ELASTODYNAMIC FMM

To exploit optimally the separation of variables $\mathbf{x}$ and $\mathbf{y}$ achieved by expansions (7), (8) and the clustering of influence terms associated with the introduction of poles $\mathbf{O}_x$, $\mathbf{O}_y$, a hierarchical oct-tree structure of elements is introduced. For that purpose, a cube of linear size d_0 containing the boundary Γ, called 'level-0 cell', is divided into eight cubes (level-1 cells), each of which is divided in the same fashion, and so on. The linear size of a level-ℓ cell is thus $d(\ell) = 2^{-\ell}d_0$. A level-ℓ cell is divided into level-$(\ell+1)$ cells unless a subdivision-stopping criterion is satisfied. Such criterion is usually based on either a preset minimum cell size D (defined relative to the shear wavelength) or a preset minimum number E of boundary elements in the cell. Such "terminal" cells are termed *leaves* As they do not feature disjoint cells, and hence do not permit the use of multipole expansions, the level-0 and level-1 subdivisions are not actually used in the computations. The level-2 subdivision, containing $4\times4\times4$ cells, is thus the highest level used in practice. A given cell may be used as a cluster of either collocation points or integration points, its centroid serving as pole $\mathbf{O}_x$ or $\mathbf{O}_y$.

The truncation parameters L_P, L_S depend upon the cell size through equation (9). Hence, in the multi-level framework, L_P, L_S are level-dependent: when applied at subdivision level ℓ, they are evaluated according to (9) with $d = d(\ell) = 2^{-\ell}d_0$.

3.1 Computation of the outgoing rays

Let C_y denote a generic leaf cell (i.e. a cell which is not subdivided into smaller cells). The *radiation functions* associated with C_y are defined by

$$\begin{aligned} F_r(\mathbf{s};C_y) &\stackrel{\text{def}}{=} FU_r - FT_r \\ F(\mathbf{s};C_y) &\stackrel{\text{def}}{=} FU - FT \end{aligned} \tag{11}$$

with

$$\begin{aligned} FU_r(\mathbf{s};C_y) &\stackrel{\text{def}}{=} k_S\mathrm{i}\int_{\Gamma\cap C_y} D_{rp}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)t_p^n(y)dy \\ FU(\mathbf{s};C_y) &\stackrel{\text{def}}{=} k_P\mathrm{i}\int_{\Gamma\cap C_y} D_{p}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)t_p^n(y)dy \\ FT_r(\mathbf{s};C_y) &\stackrel{\text{def}}{=} -k_S^2(s_q)_l E_{jkpl}\int_{\Gamma\cap C_y} n_k(y)D_{rj}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)u_p(y)dy \\ FT(\mathbf{s};C_y) &\stackrel{\text{def}}{=} -k_P^2(s_q)_l E_{jkpl}\int_{\Gamma\cap C_y} n_k(y)D_{j}(\mathbf{y}-\mathbf{O}_y,\mathbf{s}_q)u_p(y)dy \end{aligned}$$

With reference to expansions (7), (8), the radiation functions (11) are to be evaluated at quadrature points $\mathbf{s}=\mathbf{s}_q$ on the unit sphere.

Then, an *upward pass* is performed where radiation functions (11) are first computed for the lowest-level cells and then recursively aggregated by moving upward in the tree until level 2 (for which there are $4\times4\times4$ cells) is reached, a phase known as the *outer-to-outer translation*. The translation from a level-(ℓ+1) cell to its parent level-ℓ cell consists in evaluating the radiation functions at the pole $\mathbf{O}_y^\ell$ through (see Fig. 1)

$$F_r(\mathbf{s};C_y^\ell) = \sum_{C_y^{\ell+1}\in C_y^\ell} F_r(\mathbf{s};C_y^{\ell+1})\exp\Big(\mathrm{i}k_S\langle\mathbf{s},\mathbf{O}_y^\ell-\mathbf{O}_y^{\ell+1}\rangle\Big) \tag{12}$$

$$F(\mathbf{s};C_y^\ell) = \sum_{C_y^{\ell+1}\in C_y^\ell} F(\mathbf{s};C_y^{\ell+1})\exp\Big(\mathrm{i}k_P\langle\mathbf{s},\mathbf{O}_y^\ell-\mathbf{O}_y^{\ell+1}\rangle\Big) \tag{13}$$

However, the previously mentioned level-dependency through Eq. (9) of the truncation parameters L_S and L_P implies that the number and location of quadrature points on the unit sphere are also level-dependent. Application of the multipole expansions (7) and (8) at level $\ell+1$ requires that radiation functions $F_r(\mathbf{s};C_y^{\ell+1})$, $F(\mathbf{s};C_y^{\ell+1})$ be known at the level-ℓ+1 quadrature points $\mathbf{s}_q^{\ell+1}$. A subsequent *outer-to-outer translation* via (12) or (13) therefore yields the values $F_r(\mathbf{s}_q^{\ell+1};C_y^\ell)$, $F(\mathbf{s}_q^{\ell+1};C_y^\ell)$, whereas $F_r(\mathbf{s}_q^\ell;C_y^\ell)$, $F(\mathbf{s}_q^\ell;C_y^\ell)$ are needed. Evaluating the latter from the former relies on a rapid interpolation technique described in [16], which takes advantage of the uniform distribution of quadrature points along ϕ ($\phi_j=2\pi j/K,\ 0\le j\le K$). This rapid interpolation takes

advantage of the fact that radiation functions have a finite bandwidth in the Fourier space.

Let the grids of quadrature points for levels ℓ and $\ell+1$ be of the form

$$\begin{cases} (\theta_{i'}^{\ell}, \phi_{j'}^{\ell}) & 0 \le i' \le L_S^{\ell}, \quad 0 \le j' \le 2L_S^{\ell} \\ (\theta_{i}^{\ell+1}, \phi_{j}^{\ell+1}) & 0 \le i \le L_S^{\ell+1}, \quad 0 \le j \le 2L_S^{\ell+1} \end{cases}$$

with the angular coordinates chosen as explained in Section 2.3. Let $\mathcal{F}(\theta, \phi)$ denote a generic function defined on the unit sphere and having a bandwidth of $L_S^{\ell+1}$, i.e. of the form

$$\mathcal{F}(\theta, \phi) = \sum_{l=0}^{L_S^{\ell+1}} \sum_{m=-l}^{l} \mathcal{F}_{lm} Y_{lm}(\theta, \phi), \tag{14}$$

having put $L = L_S^{\ell+1}$ and where the functions $Y_{lm}(\theta, \phi)$, given by

$$Y_{lm}(\theta, \phi) = Q_l^m(\cos(\theta)) \exp(\mathrm{i}m\phi), \qquad Q_l^m(x) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(x) \tag{15}$$

in terms of the Legendre functions P_l^m, define the $L^2(S)$-orthonormal basis of spherical harmonics. The coefficients $\mathcal{F}_{lm}$ in (14) are therefore obtained by taking the scalar product (in the $L^2(S)$ sense) of $\mathcal{F}(\theta, \phi)$ with the $Y_{lm}(\theta, \phi)$. On using the numerical quadrature on S associated with level $\ell+1$, one therefore has

$$\mathcal{F}_{lm} \approx \sum_{q=1}^{Q(\ell+1)} \omega_q^{\ell+1} \bar{Y}_{lm}(\theta_i^{\ell+1}, \phi_j^{\ell+1}) \mathcal{F}(\theta_i^{\ell+1}, \phi_j^{\ell+1})$$

with indices q and (i, j) related through $\mathbf{s}^q = \mathbf{s}(\theta_i^{\ell+1}, \phi_j^{\ell+1})$ and the overbar indicating complex conjugation. On substituting the above expression into (14), the following linear relationship between the values of $\mathcal{F}$ at the quadrature points for levels ℓ and $\ell+1$ is arrived at:

$$\mathcal{F}(\theta_{i'}^{\ell}, \phi_{j'}^{\ell}) = \sum_{q=1}^{Q(\ell+1)} \omega_q^{\ell+1} \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_{lm}(\theta_{i'}^{\ell}, \phi_{j'}^{\ell}) \bar{Y}_{lm}(\theta_i^{\ell+1}, \phi_j^{\ell+1}) \mathcal{F}(\theta_i^{\ell+1}, \phi_j^{\ell+1}) \tag{16}$$

This relation provides the basis for computing radiation functions at the level-ℓ quadrature points from values at the level-$(\ell+1)$ quadrature points. On inserting (15) into (16), the following three-step procedure is obtained for computing the $\mathcal{F}(\theta_{i'}^{\ell}, \phi_{j'}^{\ell})$ from the $\mathcal{F}(\theta_i^{\ell+1}, \phi_j^{\ell+1})$, where in practice $\mathcal{F}$ is one of the radiation functions (11).

- *Step 1: direct discrete Fourier transform.* Compute the transforms $\hat{\mathcal{F}}_{im}^{\ell+1}$ de-

fined by:

$$\hat{\mathcal{F}}_{im}^{\ell+1} = \sum_{j=0}^{2L} \exp(-\mathrm{i}m\phi_j^{\ell+1})\omega_{ij}^{\ell+1}\mathcal{F}(\theta_i^{\ell+1}, \phi_j^{\ell+1})$$

- *Step 2: matrix-vector products.* Compute the $\hat{\mathcal{F}}_{i'm}^{\ell}$ defined by

$$\hat{\mathcal{F}}_{i'm}^{\ell} = \sum_{i=1}^{L} B_{i'i}^{m}\hat{\mathcal{F}}_{im}^{\ell+1}$$

  where the entries $B_{i'i}^m$ of matrix B^m are given by

$$\begin{aligned} B_{i'i}^m &= \sum_{|m|\leq l\leq L} Q_l^m(x_{i'})Q_l^m(x_i) \\ &= \sqrt{\frac{(L+1)^2-m^2}{4(L+1)^2-1}} \times \left[\frac{Q_{L+1}^m(x_{i'})Q_L^m(x_i)}{x_{i'}-x_i} - \frac{Q_{L+1}^m(x_i)Q_L^m(x_{i'})}{x_{i'}-x_i}\right] \end{aligned}$$

  having put $x_i = \cos(\theta_i^{\ell+1})$ and $x_{i'} = \cos(\theta_{i'}^{\ell})$, and with the second equality stemming from the Christoffel-Darboux formula.
- *Step 3: inverse Fourier transform.* The values $\mathcal{F}(\theta_{i'}^{\ell}, \phi_{j'}^{\ell})$ at the quadrature points for level ℓ are finally recovered by means of a discrete inverse Fourier transform:

$$\mathcal{F}(\theta_{i'}^{\ell}, \phi_{j'}^{\ell}) = \sum_{m=-L}^{L} \exp(\mathrm{i}m\phi_{j'}^{\ell})\hat{\mathcal{F}}_{i'm}^{\ell}$$

  Note that the above summation is in fact a $(2L_S^{\ell}+1)$-point discrete transform, with the missing transformed coefficients $\hat{\mathcal{F}}_{i'm}^{\ell}$ for $L = L_S^{\ell+1} < |m| \leq L_S^{\ell}$ implicitly set to zero.

The above-described scheme for transitions from a generic level $\ell+1$ to the next upper level ℓ, combining evaluation of radiation functions at level-ℓ quadrature points and the translation formulae (12) and (13), is applied until level $\ell = 2$ is reached.

3.2 Computation of the incoming rays

The radiation functions (11) synthesize, for a given cell C_y, the integral contributions of all boundary elements contained in that cell. Once computed for all non-empty cells C_y at all levels, they have to be converted into quantities defined at collocation points. The structure of the multipole expansion (5) indicates that this conversion entails a transfer from a cell C_y which supports element integrations to a (disjoint) cell C_x which supports collocation points (sometimes referred to as a *outer to inner translation*).

Let $I(C)$ denotes the *interaction list* of cell C, i.e. the list of cells of the same level as C that (a) are well-separated from C, i.e. have no point in common

(not even a single vertex) with C, and (b) are such that their parent cell is adjacent to the parent cell of C. For a cell C of level $\ell = 2$, $I(C)$ lists all cells which are well-separated from C.

The *incoming rays* at level $\ell = 2$ are defined, for a generic level-2 cell $C_x^{(2)}$, by applying the transfer functions to the level-2 radiation functions associated with all cells in $I(C_x^{(2)})$ and aggregating the result, i.e.:

$$G_r(\mathbf{s}; C_x^{(2)}) = \sum_{C_y^{(2)} \in I(C_x^{(2)})} F_r(\mathbf{s}; C_y^{(2)}) T_L(\mathbf{s}; k_S, \mathbf{O}_y^{(2)} - \mathbf{O}_x^{(2)}) \tag{17}$$

$$G(\mathbf{s}; C_x^{(2)}) = \sum_{C_y^{(2)} \in I(C_x^{(2)})} F(\mathbf{s}; C_y^{(2)}) T_L(\mathbf{s}; k_P, \mathbf{O}_y^{(2)} - \mathbf{O}_x^{(2)}) \tag{18}$$

Then, a *downward pass* is performed whereby, for $\ell \geq 2$, the incoming rays $G_r(\mathbf{s}; C_x^{\ell+1})$, $G_r(\mathbf{s}; C_x^{\ell+1})$ associated with a level-$(\ell+1)$ cell $C_x^{\ell+1}$ aggregate the contributions from (a) the level-$(\ell+1)$ cells $C_y^{\ell+1}$ belonging to the interaction list of $C_x^{\ell+1}$, and (b) the incoming rays associated with the parent cell C_x^{ℓ} with its pole translated from $\mathbf{O}_x^{\ell}$ to $\mathbf{O}_x^{\ell+1}$ (an operation sometimes referred to as *inner to inner* translation), i.e.:

$$\begin{aligned} G_r(\mathbf{s}; C_x^{\ell+1}) = \sum_{C_y^{\ell+1} \in I(C_x^{\ell+1})} & F_r(\mathbf{s}; C_y^{\ell+1}) T_L(\mathbf{s}; k_S, \mathbf{O}_y^{\ell+1} - \mathbf{O}_x^{\ell+1}) \\ & + G_r(\mathbf{s}; C_x^{\ell}) \exp\Big(\mathrm{i} k_S \langle \mathbf{s}_q, \mathbf{O}_x^{\ell+1} - \mathbf{O}_x^{\ell} \rangle\Big) \end{aligned} \tag{19}$$

$$\begin{aligned} G(\mathbf{s}; C_x^{\ell+1}) = \sum_{C_y^{\ell+1} \in I(C_x^{\ell+1})} & F(\mathbf{s}; C_y^{\ell+1}) T_L(\mathbf{s}; k_P, \mathbf{O}_y^{\ell+1} - \mathbf{O}_x^{\ell+1})) \\ & + G(\mathbf{s}; C_x^{\ell}) \exp\Big(\mathrm{i} k_S \langle \mathbf{s}_q, \mathbf{O}_x^{\ell+1} - \mathbf{O}_x^{\ell} \rangle\Big) \end{aligned} \tag{20}$$

Like with the recursive (ascending) evaluation of the radiation functions, application of equations (19) and (20) must be preceded with a procedure (known as *anterpolation*) whereby $G_r(\mathbf{s}_q^{\ell}; C_x^{\ell})$, $G(\mathbf{s}_q^{\ell}; C_x^{\ell})$ are evaluated from $G_r(\mathbf{s}_q^{\ell+1}; C_x^{\ell})$, $G(\mathbf{s}_q^{\ell+1}; C_x^{\ell})$. This step simply consists of a transposition of the interpolation scheme of section 3.1.

3.3 Final expression of BIE

Let C_x denote a generic leaf cell with center $\mathbf{O}_x$. On performing the complete upward and downward passes described in Sections 3.1 and 3.2, the contribution to the residual of integral equation (1) at all collocation points in C_x of boundary elements located in leaf cells not adjacent to C_x are accounted for,

as one has

$$\int_{y\in\Gamma\setminus A(C_x)}\Big(U^{\star}_{ip}(\mathbf{x},\mathbf{y})t^n_p(\mathbf{y})-T^{\star n}_{ip}(\mathbf{x},\mathbf{y})u_p(\mathbf{y})\Big)dy$$
$$=\frac{1}{16\pi^2\mu}\sum_{q=1}^{Q}\omega_q\Big[D_{ri}(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)G_r(\mathbf{s}_q;C_x)+\frac{k_P^2}{k_S^2}D_i(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)G(\mathbf{s}_q;C_x)\Big]$$
$$(\mathbf{x}\in C_x)\quad(21)$$

where $A(C)$ denotes the list set of cells of the same level as C that are adjacent to C (including C itself). Hence, integral equation (1) at all collocation points in C_x is recast in the form

$$C_{ip}(\mathbf{x})u_p(\mathbf{x})-\int_{y\in\Gamma\cap A(C_x)}\Big(U^{\star}_{ip}(\mathbf{x},\mathbf{y})t^n_p(\mathbf{y})-T^{\star n}_{ip}(\mathbf{x},\mathbf{y})u_p(\mathbf{y})\Big)dy$$
$$-\frac{1}{16\pi^2\mu}\sum_{q=1}^{Q}\omega_q\Big[D_{ri}(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)G_r(\mathbf{s}_q;C_x)+\frac{k_P^2}{k_S^2}D_i(\mathbf{O}_x-\mathbf{x},\mathbf{s}_q)G(\mathbf{s}_q;C_x)\Big]=0$$
$$(\mathbf{x}\in C_x)\quad(22)$$

where the near contributions, coming from integrals over elements in $\Gamma\cap A(C_x)$, are evaluated using the traditional numerical integration techniques developed for the BEM and include all singular integrals. The latter are here computed using the direct approach of [14].

3.4 Choice of FMM parameters

The accuracy and efficiency of the FMM is predicated on careful choice of two parameters. One is the truncation parameter L in the transfer function (6), which in turn influences the choice of quadrature points on the unit sphere; this has been discussed in Sec. 2.3. The other is the linear size d of a leaf cell, and is influenced by the fact that the spherical Hankel functions $h_l^{(1)}(v)$ featured in Eq. (6) behave in the large-index limit as

$$h_l^{(1)}(v)\sim\frac{C}{v}\Big(\frac{2l+1}{v}\Big)^l\qquad(l\gg v)$$

(where C is a constant) and therefore grow more than exponentially fast with l for $l>|v|$. This behavior is illustrated in Fig. 2 for several values of l. As a consequence, as discussed in [9,10], the FMM algorithm should avoid any evaluation of $h_l^{(1)}(v)$ with l significantly larger than $|v|$.

In expansion (6), the value $v=k\,|z|$ taken by the argument of $h_l^{(1)}$ is such that

$$v\geq\min k|\mathbf{O}_y-\mathbf{O}_x|$$

where $\mathbf{O}_y$ and $\mathbf{O}_x$ are centers of disjoint same-level cells. Hence, at each level, one has $|\mathbf{O}_y - \mathbf{O}_x| \geq 2d^\ell$, where d^ℓ is the size of a level-ℓ cell. So, in the worst case, one has

$$\frac{2l+1}{v} \leq \frac{2L(k)+1}{2kd^\ell} = \sqrt{3} + \frac{1 + C\log_{10}\left(\sqrt{3}kd^\ell + \pi\right)}{2kd^\ell}$$

and that upper bound should not be too large compared to unity. In particular, the size d of a leaf cell should not be a too small fraction of the wavelength, a typical range of recommended values [16,17] being $kd = O(1-2)$.

4 SYSTEM SOLUTION

The BIE formulation (22) leads to a linear system of equations of the form

$$\mathbf{A}^{\text{near}}\mathbf{x} + \mathbf{A}^{\text{far}}\mathbf{x} = \mathbf{b} \tag{23}$$

where the vector $\mathbf{x} \in \mathbb{C}^{N\times 1}$ collects all displacement and traction DOFs that are left unknown after accounting for the boundary conditions, the vector $\mathbf{b} \in \mathbb{C}^{N\times 1}$ results from the contributions of all known values of displacements and tractions in (22), and $\mathbf{A}^{\text{near}}$, $\mathbf{A}^{\text{far}} \in \mathbb{C}^{N\times N}$ are the square matrices associated with the near and far contributions involving unknown DOFs. For BEM models of large size, $\mathbf{A}^{\text{near}}$ is a sparse matrix while $\mathbf{A}^{\text{far}}$ is almost fully populated.

The FMM is inherently based on using an iterative solution technique for the set of linear equations arising from the BEM discretization. The GMRES (Generalized Minimum RESiduals [19]) algorithm is the natural, and usual, choice for this purpose. It relies on the ability to evaluate, for a given solution candidate $\mathbf{x}^{(k)}$, the *residual* $\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{A}^{\text{near}}\mathbf{x}^{(k)} - \mathbf{A}^{\text{far}}\mathbf{x}^{(k)}$ (which may be seen as the BEM-discretized version of (22)), a task requiring in turn the computation of matrix-vector products $\mathbf{A}^{\text{near}}\mathbf{x}$ and $\mathbf{A}^{\text{far}}\mathbf{x}$. As usual with iterative solvers, matrix storage is not required. In practice, storing $\mathbf{A}^{\text{far}}$ entails close to $O(N^2)$ and is thus out of the question, whereas storing $\mathbf{A}^{\text{near}}$ is a reasonable (but not mandatory) option which avoids costly reevaluations of singular and near-singular element integrals.

An important issue for algorithms such as GMRES is preconditioning. Numerical tests [20,21,22] establish that the rate of convergence of the iterative solver is strongly dependent on the spectral properties of the matrix system. The preconditioning aims at transforming the original system of equations into another, formally equivalent, one with improved spectral properties. Here, an explicit left preconditioning approach is followed, where the original sys-

tem (23) is left-multiplied by a preconditioning matrix $\mathbf{P}$, to obtain

$$\mathbf{PAx} = \mathbf{Pb} \tag{24}$$

with $\mathbf{A} = \mathbf{A}^{\text{near}} + \mathbf{A}^{\text{far}}$. The matrix $\mathbf{P}$ should ideally be such that the governing matrix of the preconditioned system (24) is the identity matrix. Of course, this would amount to solve the original system by means of a direct solver, and therefore be of little practical value. In practice, a good preconditioning strategy is such that the matrix of the preconditioned system (24) is as close as possible to the identity matrix, with computing time and storage requirements for the preconditioning significantly lower, and growing slower with N, than those expected for the original system of equations. Here, following [23] and earlier works from the computational electromagnetism community [24], the preconditioning matrix $\mathbf{P}$ is chosen as the SParse Approximate Inverse (SPAI) of $\mathbf{A}$. A complete description of this approach is given in [25,26], where the non-singular character and other theoretical properties of the SPAI preconditioner are discussed.

In its simplest version, a SPAI $\mathbf{P}$ of $\mathbf{A}$ is defined by choosing *a priori* a sparsity pattern for $\mathbf{P}$, and then finding $\mathbf{P}$ as the solution to the minimization problem

$$\min \|\mathbf{I} - \mathbf{PA}\|^2 \tag{25}$$

(where $\|\cdot\|$ denotes the Frobenius matrix norm) which satisfies the preset sparsity pattern. The minimization problem (25) is then shown to take the form

$$\min \left\|\mathbf{I} - \mathbf{P}^l\mathbf{A}\right\|^2 = \sum_{k=1}^{n} \min \left\|e_k - P_k^l\mathbf{A}\right\|^2 \tag{26}$$

where e_k and P_k^l are the k-th row of the identity matrix and matrix $\mathbf{P}^l$ respectively. Hence, each row P_k^l of $\mathbf{P}$ can be found by solving an independent least-squares problem whose size is equal to the number of nonzero entries in $\mathbf{P}^l$ predicted by the preset sparsity pattern. The sparsity pattern of $\mathbf{P}^l$ should ideally be chosen such that the entries of $\mathbf{A}^{-1}$ with largest moduli are retained. However, the locations of these entries are not known *a priori*. In [25] the sparsity pattern is found for every row (or column) with an adaptative strategy. The positions of the non-zero terms in $\mathbf{P}^l$ are iteratively modified to minimize the residual provided by Eq. (26). In [24] a different approach is adopted, where four heuristics are proposed for choosing a fixed sparsity pattern for $\mathbf{P}$. Here, the sparsity pattern of $\mathbf{P}$ is defined as corresponding to the location of the m largest entries in each row of $\mathbf{A}$ (with $m \ll n$). On defining the sparse matrix $\hat{\mathbf{A}}$ from $\mathbf{A}$ according to this same sparsity pattern, the minimization problem (26) can be simplified by dropping out the zero rows, to obtain:

$$\sum_{k=1}^{n} \min \left\|\hat{e}_k - \hat{P}_k^l\hat{\mathbf{A}}\right\|^2 \tag{27}$$

where $\hat{(\cdot)}$ denote entities reduced according to the sparsity pattern. Each least squares problem in (27) is therefore of small size $m \ll n$ (and hence can be solved inexpensively). Well-posedness of the $m \times m$ least squares problems is assured by including all diagonal entries of $\mathbf{A}$ in the sparsity pattern.

This type of preconditioners seems to be quite effective also for non-symmetric matrices even if a sparsity of around $97 - 99\%$ is imposed for $\mathbf{P}^l$.

The stopping criterion adopted for the GMRES algorithm is based on the norm of the residual of the original system of equations, according to

$$\frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}\|}{\|\mathbf{b}\|} \leq \text{Tol}$$

where the tolerance Tol is chosen *a priori*.

5 NUMERICAL EXAMPLES

Numerical results obtained on the basis of the present FM-BEM are now presented. The convergence rate of the iterative system solution GMRES was accelerated by using the left-preconditioning strategy of Section 4. The efficiency (in CPU time sense) and the accuracy of results is compared to the classical BEM, for which GMRES with left-preconditioning was also used. The truncation parameters were set according to (10), with $C = 7.5$ in (9) found to provide the best compromise between efficiency and accuracy. The recursive cell subdivision is stopped at the level for which the cell size d is such that

$$k_S d_{min} \leq \frac{2\pi}{3}$$

The GMRES algorithm is used with a stopping criterion defined by

$$\frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}\|}{\|\mathbf{b}\|} \leq 10^{-3}$$

and a restart parameter of 600 iterations. The problems were run on a work station with 4 Processors 64bits Opteron 2.2 Ghz and 16 GB of core memory.

5.1 Cubic box under traction

A cube with dimensions $H \times H \times H$ (Fig. 3) is considered. Normal displacements and in-plane tractions are set to zero on faces $x, z = \pm H/2$ and $y = -H$, while a time-harmonic normal traction p is imposed on the face $y = 0$. The

relevant parameters are set as follows: $H = 1\,\mathrm{m}$, $\mu = 10^{11}\,\mathrm{N/m^2}$, $\nu = 0.25$, $p = 100\,\mathrm{N/m^2}$, $\rho = 8\,10^3\,\mathrm{Kg/m^3}$.

BE meshes made of four-noded bilinear elements are set up, so that each face features a uniform grid of $M \times M$ four-noded bilinear elements of size $a \times a$. The linear element size a is chosen according to the frequency so that the shear wavelength is $\lambda_T = 20a$. Four cases with increasing frequencies and problem sizes have been considered, with meshes such that each face of the cube features $M \times M$ square elements, with $M = 20, 34, 48, 86$ respectively. The resulting numbers of BE DOFs are indicated in Table 1, and the mesh for problem 1 shown in Fig. 4. The oct-trees featured in problems 1, 2, 3 and 4 feature 2, 3, 3 and 4 levels, respectively.

CPU times and GMRES iteration counts recorded for the various phases of the computation are shown in Table 2, where the heading "other" indicates the cumulative CPU time entailed by all tasks other than the computation of incoming and outgoing rays. Figure 5 graphically shows that the CPU times recorded for the present FM-BEM approach are significantly lower than those needed for the traditional BEM on the same problems. Problem 4, involving $O(10^5)$ DOFs, could not be solved using traditional BEM because the size of the resulting governing fully-populated BEM matrix (about 250 GB) largely exceeds the RAM capacity of the computer on which the computations were run. Moreover, the CPU times per GMRES iteration are consistent with the theoretical estimate $O(N \ln N)$ [10], as shown in Fig. 6.

In Fig. 7, the y-component of displacement results along the s-coordinate defined in Fig. 3), obtained with the present FM-BEM and the traditional BEM are seen to agree well for the four above-defined problems. The number of GMRES iterations is seen to increase with the frequency (to which the problem size is adjusted), as expected since the spectral properties of the linear system being solved is frequency-dependent. The preconditioning strategy presented in section 4 leads to a substantial reduction of the GMRES iteration count, as can be seen in Table 2, where the iteration counts recorded with and without preconditioning are given. For completeness and comparison purposes, the GMRES iteration counts and overall CPU times recorded using the traditional BEM with preconditioning are given in Table 3.

5.2 Ellipsoidal cavity in a rectangular box

A traction-free ellipsoidal cavity (with principal semiaxes $(a_1, a_2, a_3) = (3, 2, 2)a$, a being an arbitrary reference length, and axis a_1 oriented along direction $(1, 1, 1)$) embedded in a rectangular box (with dimensions $(15a, 15a, 20a)$ in directions (x, y, z), respectively, see Fig. 8) is considered. The material param-

eters and the boundary conditions on the external surface are defined as in the first example. The frequency is set so that $\lambda_T = 5a$. The cavity center and the box center coincide. Four-noded elements are used for the cavity boundary (Fig. 8) and the box surface, the mesh densities being adjusted so as to feature 20 and 30 elements per shear wavelength on the external and cavity surface, respectively. Overall, the problem involves $N = 81054$ BE DOFs. Its solution required 441 iterations and 29714 seconds. The y-component of the displacement on the box surface is shown in Fig. 9. This example corresponds to a generic forward simulation for a defect identification problem.

6 CONCLUSIONS

A fast multipole accelerated BEM formulation for solving 3-D frequency domain elastodynamics has been presented in this paper. The expansion of the elastodynamic kernels is based on the diagonal-form, plane-vave, expansion of the scalar acoustic full-space fundamental solution. The divergent behaviour in the large-index limit of the spherical Hankel function of the first kind featured in that expansion led to the imposition of a minimum size (relative to the wavelength) for the leaf cells. Another important parameter, namely the truncation parameter of the transfer function, is determined as a function of the cell size (and hence the level) by means of a rule previously introduced for time-harmonic acoustic or electromagnetic computations.

Numerical results show that the numerical solution obtained by the present elastodynamic FM-BEM is, in practical terms, identical to that obtained by the traditional BEM (the two solutions differ by at most 3%) while requiring about 10 times less CPU time for the examples run (which were of size $N = O(10^4 - 10^5)$. The convergence rate of the GMRES iterative solver was found to be significantly accelerated by left-preconditioning the linear system with the sparse approximate inverse (SPAI) technique.

ACKNOWLEDGEMENTS

This work was supported by the Ministerio de Educación y Ciencia of Spain. (Project: DPI2004-08147-C02-02). The financial support is gratefully acknowledged. We also acknowledge Dr. Massimiliano Margonari for having kindly provided his elastostatic FM-BEM source code.

References

[1] Rokhlin V. Rapid solution of integral equations of classical potential theory. J Comput Phys 1985; 60:187–207.

[2] Greengard L, Rokhlin V. A fast algorithm for particle simulations. J Comput Phys 1987; 73:325–348.

[3] Lu CC, Chew WC. Fast algorithm for solving hybrid integral equations. IEEE Proc H 1993; 140:455–460.

[4] Lu CC, Chew WC. A multilevel algorithm for solving a boundary integral equation of wave scattering. Microw Opt Technol Lett 1994; 7:466–470.

[5] Nishimura N. Fast multipole accelerated boundary integral equation methods. Appl Mech Rev 2002; 55:299–324.

[6] Takahashi T, Nishimura N, Kobayashi S. A fast BIEM for three-dimensional elastodynamics in time domain. Eng Anal Bound Elem 2003; 27:491–506.

[7] Fujiwara H. The fast multipole method for solving integral equations of three-dimensional topography and basin problems. Geophys J Int 2000; 140:198–210.

[8] Messiah A. Quantum mechanics. New York: John Wiley & Sons; 1968.

[9] Darve E, Havé P. A fast multipole method for Maxwell equations stable at all frequencies. Philos Trans R Soc A-Math Phys Eng Sci 2004; 362:603–628.

[10] Darve E. The fast multipole method: numerical implementation. J Comput Phys 2000; 160:195–240.

[11] Greengard L, Rokhlin V. A new version of the fast multipole method for the Laplace equation in three dimensions. Acta Num 1997; 6:229–269.

[12] Bonnet M. Boundary integral equation methods for solids and fluids. Wiley; 1995.

[13] Mantic V. A new formula for the C-matrix in the Somigliana identity. J Elast 1993; 33:191–201.

[14] Giuggiani M, Gigante A. A general algorithm for multidimensional Cauchy principal value integrals in the boundary element method. J App Mech-Trans ASME 1990; 57:906–915.

[15] Abramowitz M, Stegun IA Editors. Handbookof mathematical functions with formulas, graphs, and mathematical tables. New York: Dover Publications Inc.; 1992 (Reprint of the 1972 edition).

[16] Sylvand G. La méthode multipôle rapide en électromagnétisme: performances, parallélisation, applications. PhD thesis, École des ponts et chaussées, Paris, France; June 2002.

[17] Carpentieri B, Duff IS, Giraud L, Sylvand G. Combining fast multipole techniques and an approximate inverse preconditioner for large electromagnetism calculations. SIAM J Sci Comput 2005; 27:774–792.

[18] Ohnuki S, Chew WC. Numerical accuracy of multipole expansion for 2D MLFMA. IEEE T Antenn Propag 2003; 51:1883–1890.

[19] Saad Y, Schultz MH. Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Comput 1986; 7:856–869.

[20] Mansur WJ, Araujo FC, Malaghini JEB. Solution of BEM systems of equations via iterative techniques. Int J Numer Methods Eng 1992; 33:1823–1841.

[21] Kane JH, Keyes DE, Prasad KG. Iterative solution techniques in boundary element analysis. Int J Numer Methods Eng 1991; 31:1511–1536.

[22] Leung CY, Walker SP. Iterative solution of large three-dimensional BEM elastostatics analyses using the GMRES technique. Int J Numer Methods Eng 1997; 40:2227–2236.

[23] Margonari M, Bonnet M. Fast multipole method applied to elastostatic BEM-FEM coupling. Comput Struct 2005; 83:700–717.

[24] Alléon G, Benzi M, Giraud L. Sparse approximate inverse preconditioning for dense linear system arising in computational electromagnetics. Num Algorithms 1997; 16:1–15.

[25] Grote MJ, Huckle T. Parallel preconditioning with sparse approximate inverses. SIAM J Sci Comput 1997; 18:838–853.

[26] Benzi M, Tuma M. A sparse approximate inverse preconditioner for nonsymmetric linear sytem. SIAM J Sci Comput 1998; 19:968–994.

List of Figures

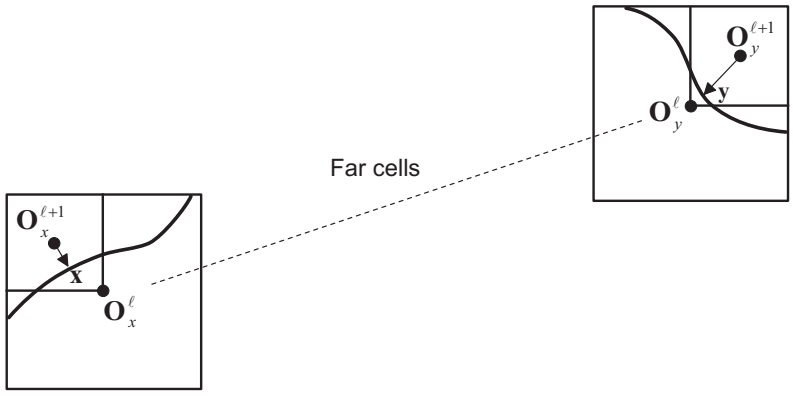


Fig. 1. Pole translations.

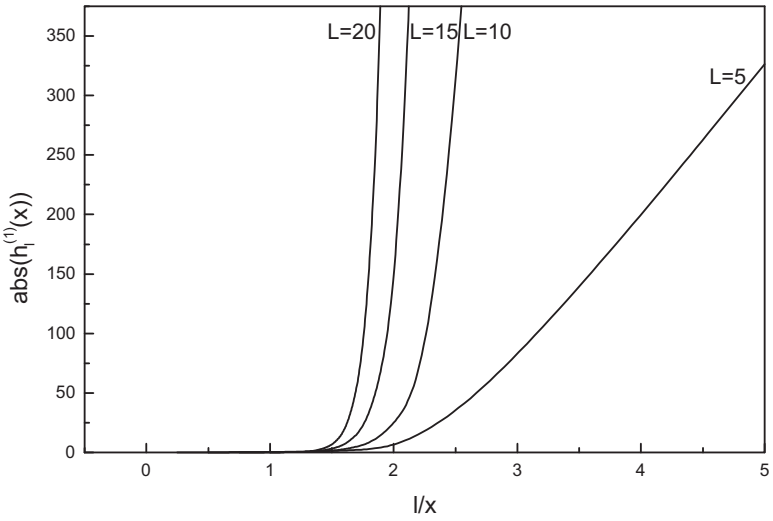


Fig. 2. Divergence of spherical Hankel functions $h_l^{(1)}(x)$ for $l/|x| > 1$.

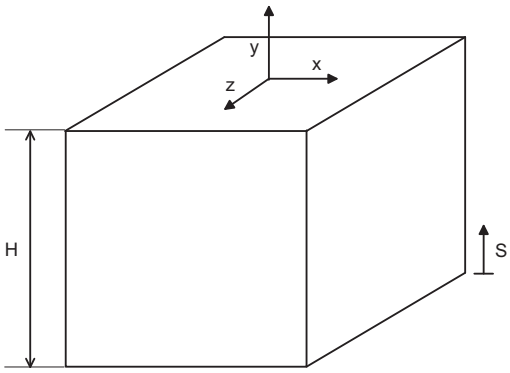


Fig. 3. Cube under traction: geometry and notations.

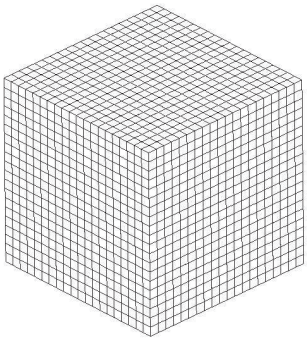


Fig. 4. Cube under traction: mesh for problem 1 (2400 elements).

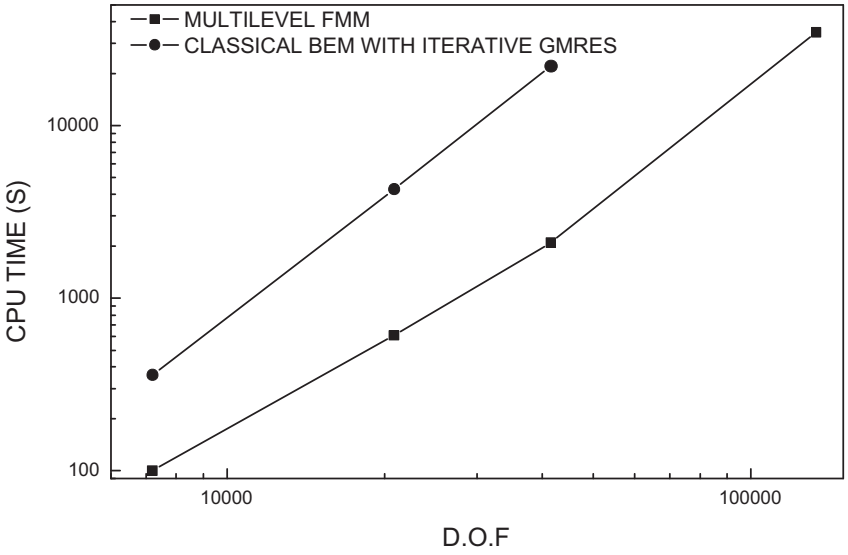


Fig. 5. Cube under traction: overall CPU times for the FM-BEM and classical BEM computations.

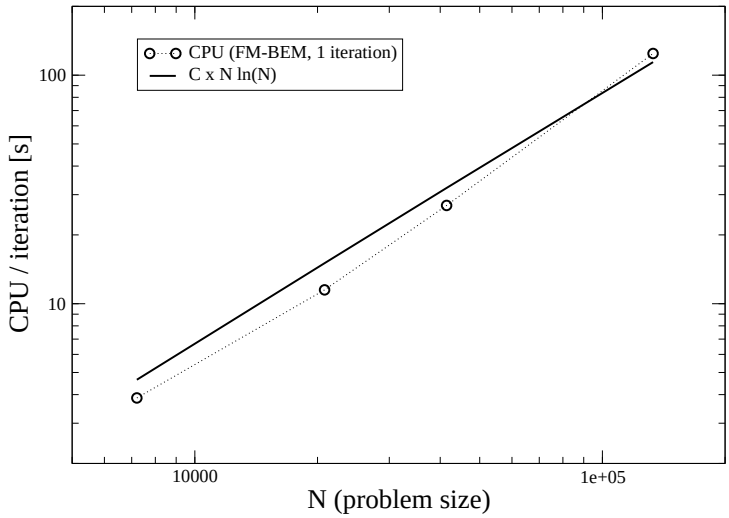


Fig. 6. Cube under traction: CPU times per iteration for the FM-BEM, and best fit of the form CPU=$C \times N \ln N$, with $C \approx 7.92 \cdot 10^{-5} s$.

List of Tables

Table 1
Cube under traction: frequencies and problem sizes.

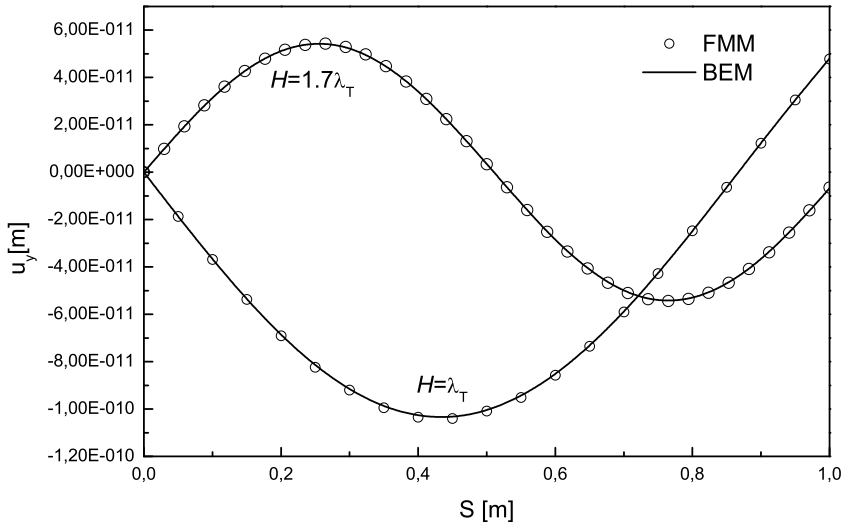
Problem	H/λ_T	problem size N
1	1	7206
2	1.7	20814
3	2.4	41478
4	4.3	133134

Table 2
Cube under traction: CPU times (seconds) and GMRES iteration counts both with and without SPAI preconditioning (Pre and No-Pre, respectively).

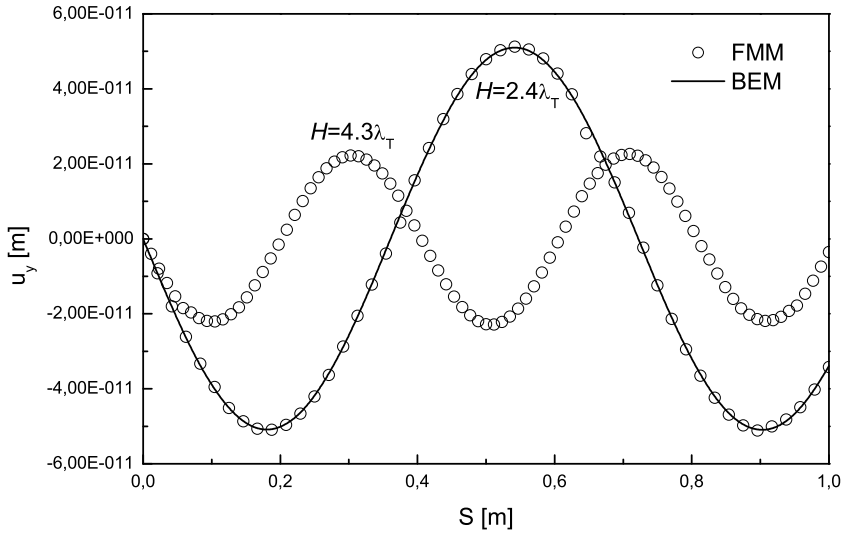
Problem	Out-rays	In-rays	Other	Iters (Pre)	Iters (No-Pre)	Total time (Pre)
1	2.3	1.2	0.03	26	32	100.5
2	4.4	6.5	0.05	53	79	608.9
3	13.2	12.8	0.2	78	130	2098.1
4	46.3	74.8	1.9	279	685	34708.1

Table 3
Cube under traction: CPU times (seconds) and GMRES iteration counts using the traditional BEM approach.

Problem	Iters (Pre)	Total time (Pre)
1	24	359.1
2	44	4270.2
3	69	22135.8
4	—	—



(a) y-component of displacement along s-coordinate (see fig. 3) for cube under traction (problems 1 and 2)



(b) y-component of displacement along s-coordinate (see fig. 3) for cube under traction (problems 3 and 4)

Fig. 7. Cube under traction: results.

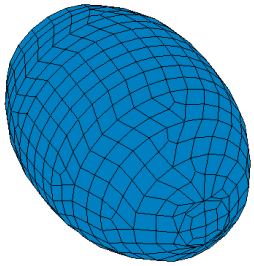


Fig. 8. Ellipsoidal cavity in a rectangular box: mesh of the cavity boundary. Overall problem size: $N = 81054$.

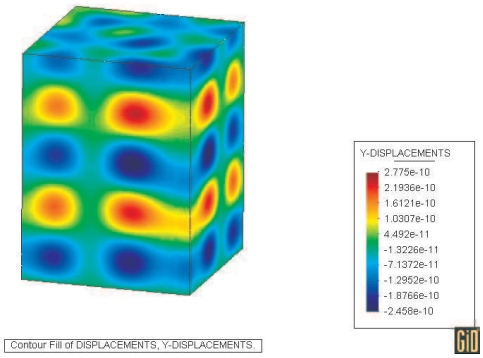


Fig. 9. Ellipsoidal cavity in a rectangular box: y-component of displacement [m] on the box surface.